

CV assignment 2 by Rishabh Bajpai and Jagriti Agnihotri

Introduction- The whole problem is considered as a binary classification problem and HOG features vectors are used in the classification of images.

Dataset Used- The dataset contains 1400 images, 700 of both aeroplanes and lakes. Out of which 66% of images are taken as training images and rest for testing. Aeroplanes images are considered as class 0 and lakes images are considered as class 1.

Code- There are two python files main.py and function.py. Python 3 is used.

The function.py contains the functions and logic for getting the HOG feature vector of images.

Functions:	Use:
preprocess(filename)	Preprocess input image array by -Conversion of unit8 encoding to int16 so that it can take negative values as well -resizing the image -normalize the contrast of the image -returns imagePre (preprocessed image)
cal_gradient(imagePre)	Compute the gradient of input image Returns g (gradient image array) ,g_angle (image gradient angle array)
calc_histPos(value,value_grad,histogram_cell)	Stores the gradient in the gradient histogram (histogram_cell) according to its proportion Input-value is gradient angle's value Value_grad is gradient's value Histogram_cell is histogram of a cell Output- Updated Histogram_cell
calc_histogram(g,ga)	It computes histogram of a cell from input g, and ga Returns updated histogram_cell_mat (matrix of histograms)
hist_gradient(gradient,gradient_angle)	Computes histogram matrix (histogram_block_mat)
normalized_hist(histogram_block)	Normalize the given histogram block matrix
hist_vector(histogram_block)	Reshape the histogram block matrix into feature vector of that block

Block_norm(histogram_block_mat)	Return Normalizes block histogram matrix
hog(filename)	Compute hog feature vector from input image's path

The main.py file contains the code for the formation of test and training dataset and the HOG vector is generated using function.py file and then feature matrix is created with suitable labels i.e 0 for aeroplane and 1 for lake image. Then for classification SVM classifier is used.

Results- Accuracy is 97%.

Accuracy: 0.9705882352941176				
	precision	recall	f1-score	support
0.0	1.00	0.94	0.97	252
1.0	0.94	1.00	0.97	224
avg / total	0.97	0.97	0.97	476